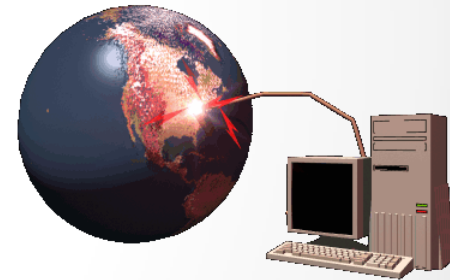




S1 Informatic Engineering

Advanced Software Engineering

Real Time System



By:

Egia Rosi Subhiyakto, M.Kom, M.CS

Informatic Engineering Department

egia@dsn.dinus.ac.id

+6285640392988

SYLLABUS

8. Web App. Process and Architecture

9. WebE Design (1)

10. WebE Design (2)

11. Real Time System

12. Testing Web App.

13. Present Final Project

14. Present Final Project

Objectives

- To explain the concept of a real-time system and why these systems are usually implemented as concurrent processes
- To describe a design process for real-time systems
- To explain the role of a real-time operating system
- To introduce generic process architectures for monitoring and control and data acquisition systems

Real-time systems

- Systems which monitor and control their environment.
- Inevitably associated with hardware devices
 - Sensors: Collect data from the system environment;
 - Actuators: Change (in some way) the system's environment;
- Time is critical. Real-time systems **MUST** respond within specified times.

Definition

- A **real-time system** is a software system where the correct functioning of the system depends on the results produced by the system and the time at which these results are produced.
- A **soft real-time system** is a system whose operation is degraded if results are not produced according to the specified timing requirements.
- A **hard real-time system** is a system whose operation is incorrect if results are not produced according to the timing specification.



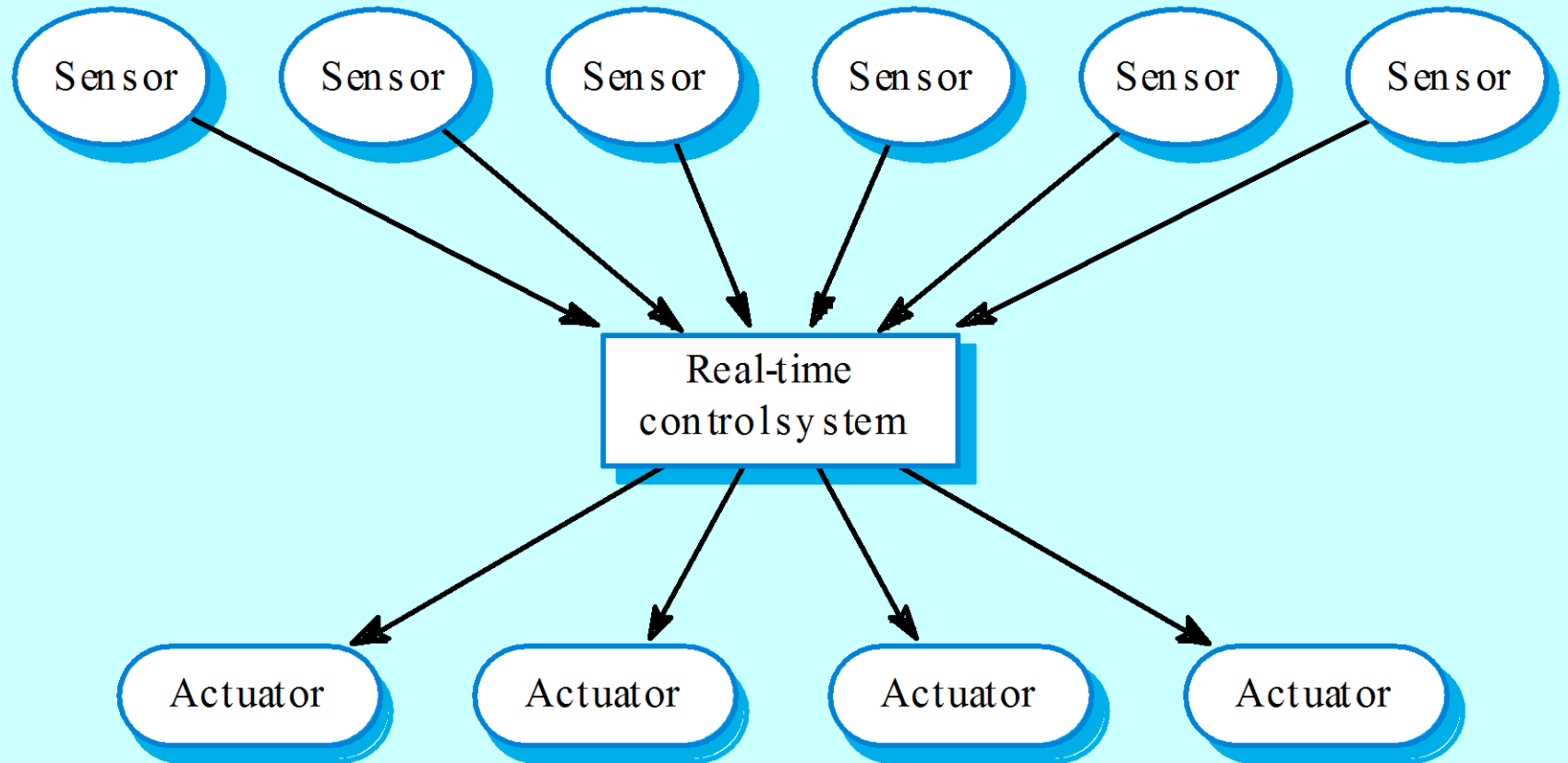
Stimulus/Response Systems

- Given a stimulus, the system must produce a response within a specified time.
- **Periodic stimuli.** Stimuli which occur at predictable time intervals.
 - For example, a temperature sensor may be polled 10 times per second.
- **A periodic stimuli.** Stimuli which occur at unpredictable times
 - For example, a system power failure may trigger an interrupt which must be processed by the system.

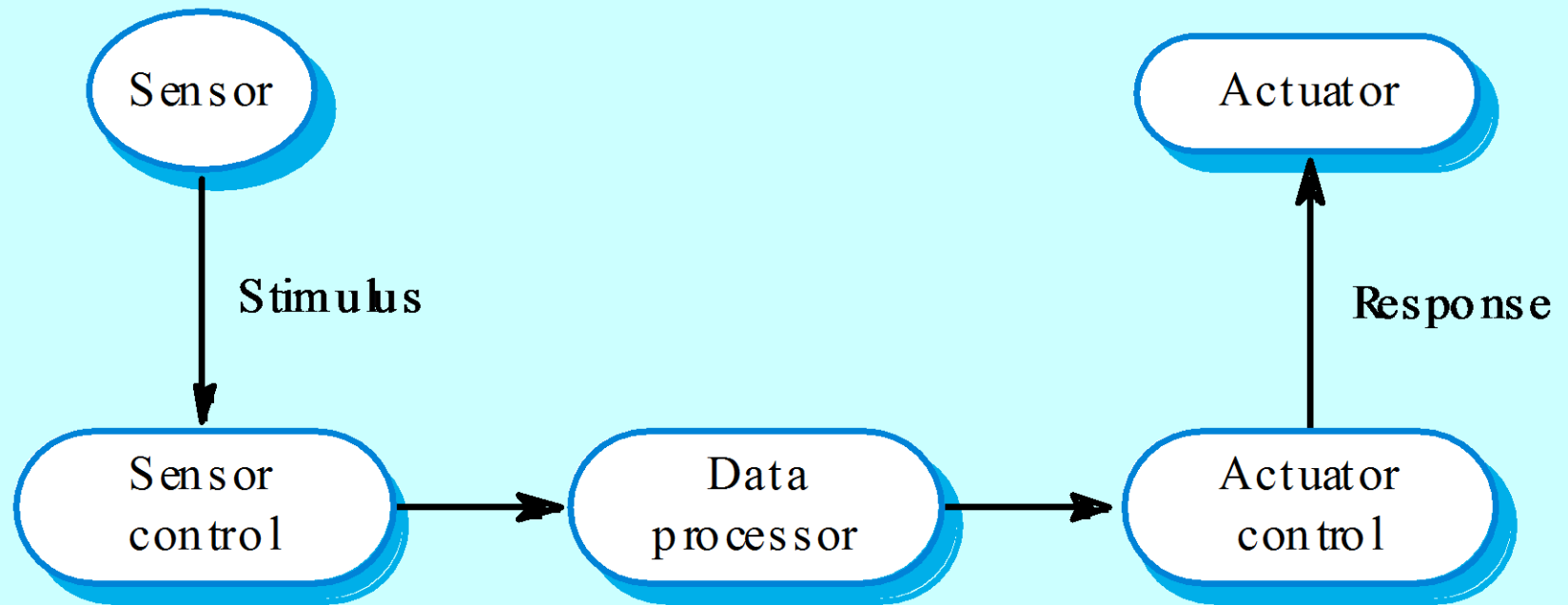
Architectural considerations

- Because of the need to respond to timing demands made by different stimuli/responses, the system architecture must allow for fast switching between stimulus handlers.
- Timing demands of different stimuli are different so a simple sequential loop is not usually adequate.
- Real-time systems are therefore usually designed as cooperating processes with a real-time executive controlling these processes.

A real-time system model



Sensor/actuator processes



System elements

- Sensor control processes
 - Collect information from sensors. May buffer information collected in response to a sensor stimulus.
- Data processor
 - Carries out processing of collected information and computes the system response.
- Actuator control processes
 - Generates control signals for the actuators.

Real-time programming

- Hard-real time systems may have to be programmed in assembly language to ensure that deadlines are met.
- Languages such as C allow efficient programs to be written but do not have constructs to support concurrency or shared resource management.
- Java is not really suitable for real-time programming as it does not allow the programmer to control timing. However, a number of real-time Java variants have been proposed.

Real Time Characteristics

- ✓ **Timelines** - time is an important part. Real-time systems must respond within the specified time.
- ✓ **Dynamic Internal Structure** - Keep system components that can perform dynamic reconfiguration to adapt to external environmental conditions.

Real Time Characteristics

- ✓ **Reactiveness** - continuously conditions may respond to a variety of events that can not be predicted arrival time sequence.
- ✓ **Concurrency** - support aspects of synchronous, asynchronous, communication, interrupt handling is an important part in solving this concurrency

R-T systems design process

- Identify the stimuli to be processed and the required responses to these stimuli.
- For each stimulus and response, identify the timing constraints.
- Aggregate the stimulus and response processing into concurrent processes. A process may be associated with each class of stimulus and response.

R-T systems design process

- Design algorithms to process each class of stimulus and response. These must meet the given timing requirements.
- Design a scheduling system which will ensure that processes are started in time to meet their deadlines.
- Integrate using a real-time operating system.

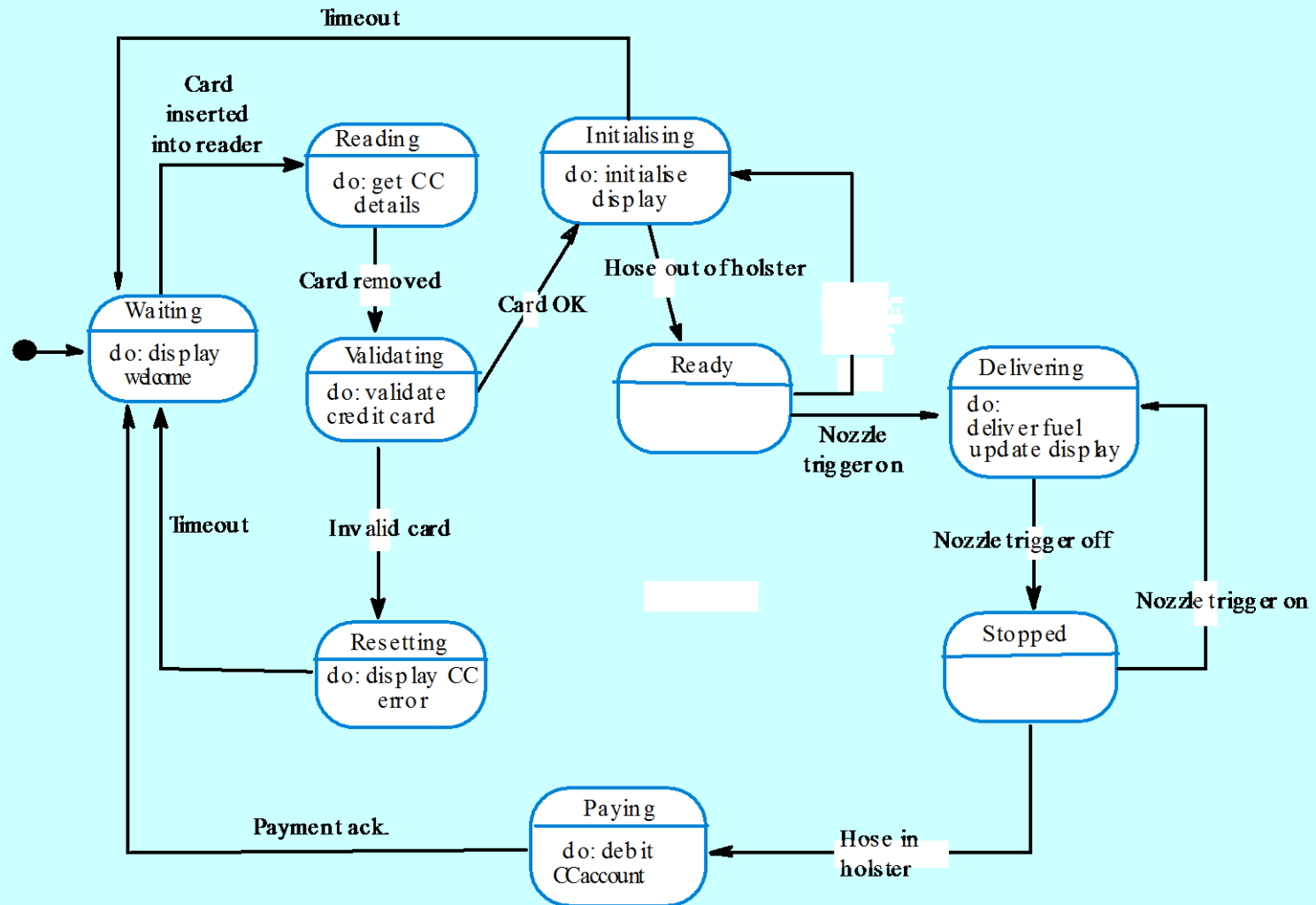
Timing constraints

- May require extensive simulation and experiment to ensure that these are met by the system.
- May mean that certain design strategies such as object-oriented design cannot be used because of the additional overhead involved.
- May mean that low-level programming language features have to be used for performance reasons.

Real-time system modelling

- The effect of a stimulus in a real-time system may trigger a transition from one state to another.
- Finite state machines can be used for modelling real-time systems.
- However, FSM models lack structure. Even simple systems can have a complex model.
- The UML includes notations for defining state machine models

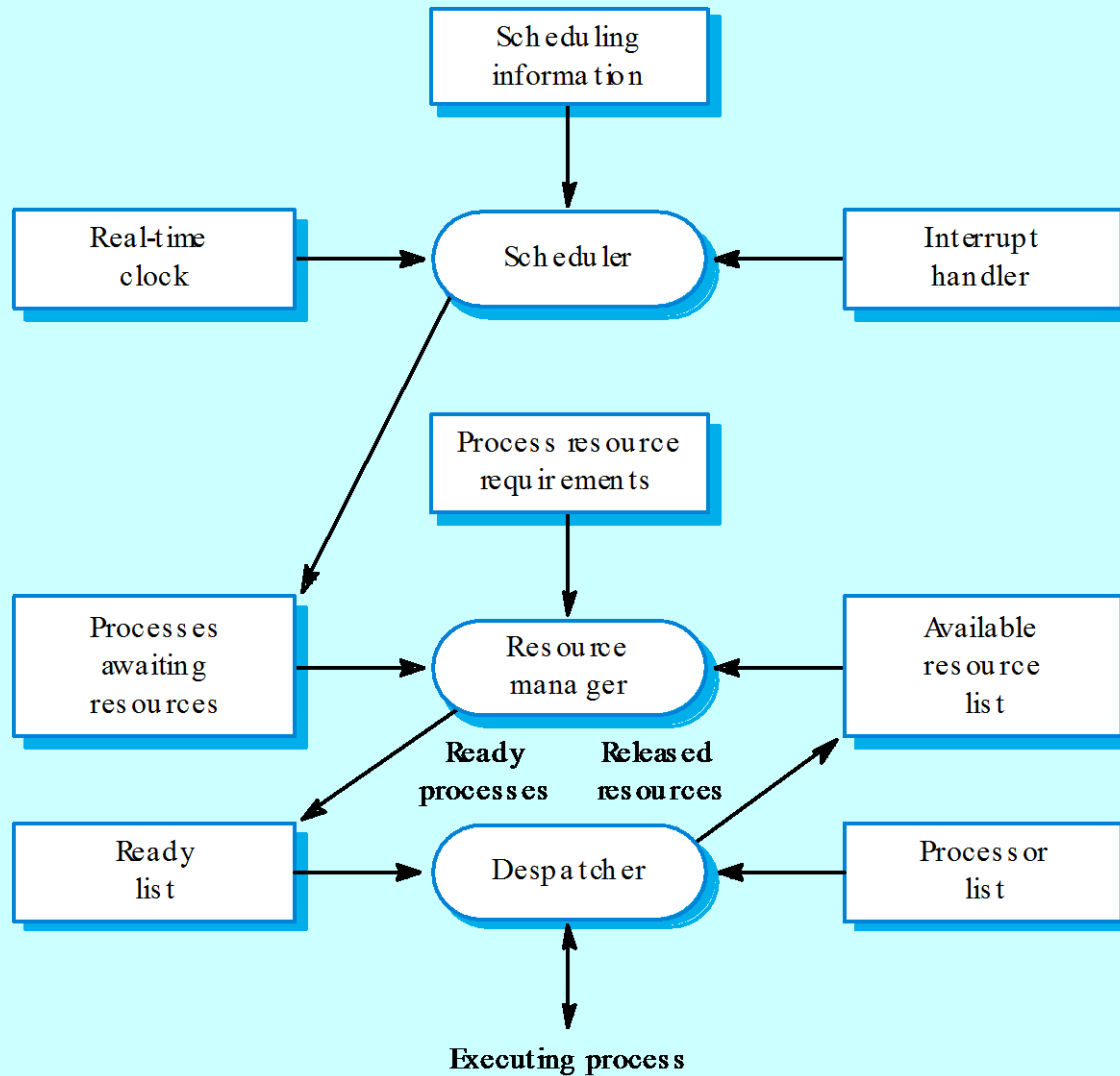
Petrol pump state model



Real-time operating systems

- Real-time operating systems are specialised operating systems which manage the processes in the RTS.
- Responsible for process management and resource (processor and memory) allocation.
- May be based on a standard kernel which is used unchanged or modified for a particular application.
- Do not normally include facilities such as file management.

Real-time OS components



Real-time OS components

Real-time clock

- Provides information for scheduling process.

Interrupt handler

- Manage certain periodic requests for service

Scheduler

- Choosing the next process to be run.

Resource manager

- Allocate memory and processor resources.

Dispatcher

- Begin the process of execution

Process priority

- The processing of some types of stimuli must sometimes take priority.
- Interrupt level priority. Highest priority which is allocated to processes requiring a very fast response.
- Clock level priority. Allocated to periodic processes.
- Within these, further levels of priority may be assigned.

Interrupt servicing

- Control is transferred automatically to a pre-determined memory location.
- This location contains an instruction to jump to an interrupt service routine.
- Further interrupts are disabled, the interrupt serviced and control returned to the interrupted process.
- Interrupt service routines **MUST** be short, simple and fast.

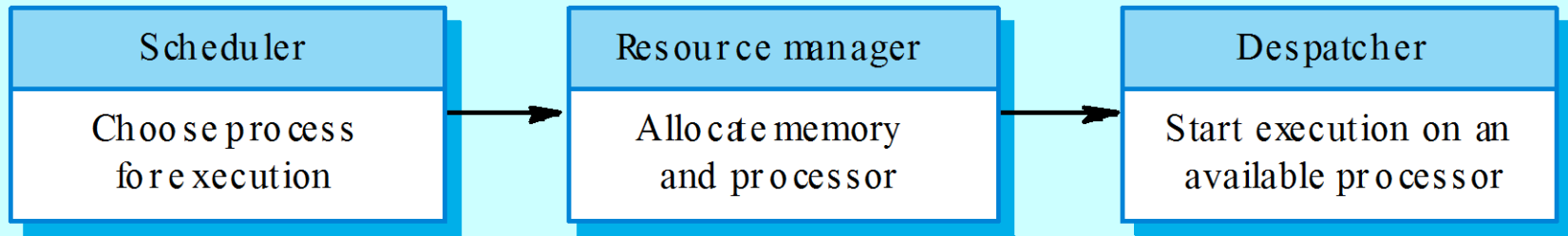
Periodic process servicing

- In most real-time systems, there will be several classes of periodic process, each with different periods (the time between executions), execution times and deadlines (the time by which processing must be completed).
- The real-time clock ticks periodically and each tick causes an interrupt which schedules the process manager for periodic processes.
- The process manager selects a process which is ready for execution.

Process management

- Concerned with managing the set of concurrent processes.
- Periodic processes are executed at pre-specified time intervals.
- The RTOS uses the real-time clock to determine when to execute a process taking into account:
 - Process period - time between executions.
 - Process deadline - the time by which processing must be complete.

RTE process management



Key points

- Real-time system correctness depends not just on what the system does but also on how fast it reacts.
- A general RT system model involves associating processes with sensors and actuators.
- Real-time systems architectures are usually designed as a number of concurrent processes.
- Real-time operating systems are responsible for process and resource management.

Next Week – December 17th, 2014

Caused in next week I have FGD (Forum Group Discussion) with Dishubkominfo Kota Semarang, please follow this instruction:

1. Read Material about Testing Web Application (Download in Siadin) – One of material for Final Exam
2. Complete your report (Design Report), Print and collected in my room/ locker + Absence
3. For last meeting (December 24th, 2014) present your final project, for sequence presentation, I will be randomized today

Presentation Outline

Please Put into your Slide :

1. Purpose of Website
2. Scope of Problem
3. References
4. General Description of Website
5. Software Function (Website Function)
6. Characteristic of Users
7. Restriction
8. Use Case Diagram
9. Class Diagram
10. Decomposition Logic Software
11. Data Design (Database) – For Each Table
12. Show Main Display Design Website

→ Present Application (Website)

➤ Time to presentation: 35 minutes for 1 group including QA

THANK YOU